**ADC Initialization**

void initADC(void)

{

*// Setup ADC0 configuration.*

    SysCtlPeripheralEnable(SYSCTL\_PERIPH\_ADC0);

*// Enable sample sequence 3 with a processor signal trigger.  Sequence 3*

*// will do a single sample when the processor sends a signal to start the conversion*

    ADCSequenceConfigure(ADC0\_BASE, 3, ADC\_TRIGGER\_PROCESSOR, 0);

*// Configure step 0 on sequence 3.  Sample channel 0 (ADC\_CTL\_CH0) in*

*// single-ended mode (default) and configure the interrupt flag*

*// (ADC\_CTL\_IE) to be set when the sample is done.  Tell the ADC logic*

*// that this is the last conversion on sequence 3 (ADC\_CTL\_END).  Sequence*

*// 3 has only one programmable step.  Sequence 1 and 2 have 4 steps, and*

*// sequence 0 has 8 programmable steps.  Since we are only doing a single*

*// conversion using sequence 3 we will only configure step 0.*

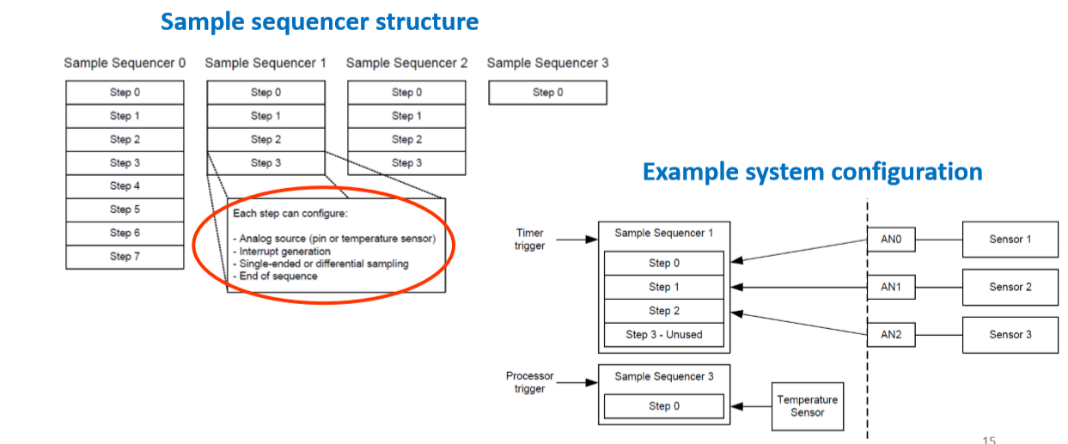
    ADCSequenceStepConfigure(ADC0\_BASE, 3, 0, ADC\_CTL\_CH0 | ADC\_CTL\_IE | ADC\_CTL\_END);

*// Enable sample sequence*

    ADCSequenceEnable(ADC0\_BASE, 3);

}

1. Setup ADC0 configuration
2. Enable sample sequence 3 with a  processor signal trigger
3. Configure step 0 on sequence 3 (changed for the lab **ADC\_CTL\_CH0** to **ADC\_CTL\_CH9**) this changes analogue pin.
4. Enable sample sequence
5. … add Interrupts



**12-bit ADC Module**

* 2 ADC modules

Each have = 4 sample sequencers - Each sequencer with configurable trigger - Each sequencer has 1, 4 or 8 steps

* 8 input channels and an internal temperature sensor.